

Huiswerk kwartaal 3 week 5

Deze week ga je twee verschillende manieren leren om een digitaal filter te ontwerpen:

- met behulp van de Inverse Discrete-Time Fourier Transform (IDTFT);
- door het plaatsen van polen en nulpunten in het Z-domein.

Volgende week komt daar nog een derde methode bij.

Na afloop van deze week:

- kun je een digitaal filter ontwerpen met behulp van de Inverse Discrete-Time Fourier Transform (IDTFT);
- ken je het verschil tussen een Finite Impulse Response (FIR)-filter en een Infinite Impulse Response (IIR)-filter;
- kun je de overdracht van een LTI-systeem ook omzetten naar het Z-domein m.b.v. de Z-transformatie;
- ken je de belangrijkste eigenschappen van deze transformatie (lineariteit en tijdverschuiving);
- ken je de relatie tussen de discrete Fouriertransformatie en de Z-transformatie en dus ook de relatie tussen Ω en z ;
- ken je het effect van polen en nulpunten in het Z-domein;
- snap je dat complexe polen of nulpunten alleen in complex-toegevoegde paren voorkomen;
- kun je een digitaal filter ontwerpen door het plaatsen van polen en nulpunten in het Z-domein;
- kun je een overdracht in het Z-domein terugtransformeren naar het tijddomein met de inverse Z-transformatie.

Les 1

De les begint met een terugblik op de stof van [week 3](#). Zie [sheet 2](#) tot en met [8](#) van de presentatie. Om te kijken of je de stof van week 3 nog goed beheerst, kun je de [oefenopgaven 3.5.1 en 3.5.2](#) maken. De [antwoorden](#) en [uitwerkingen](#) staan op de wiki.

De eerste manier om een digitaal filter te ontwerpen is gegeven op [sheet 10 van de presentatie](#). We definiëren simpelweg de gewenste $H(\Omega)$ en transformeren die naar $h[n]$ met de IDTFT.

Deze $h[n]$ implementeren we vervolgens als LTI-systeem, zodat de filterbewerking real-time uitgevoerd kan worden in het tijddomein.

Op [sheet 11](#) is een voorbeeld gegeven van een perfect laagdoorlaatfilter met een genormeerde kantel-hoekfrequentie Ω_c van $\frac{2}{5}\pi$ wat overeenkomt een kantelfrequentie f_c van $\frac{1}{5}f_s$. Deze overdracht kunnen we transformeren naar het tijddomein met de IDTFT:

$$h[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} H(\Omega) e^{j\Omega n} d\Omega \quad (1)$$

Omdat het spectrum $H(\Omega)$ periodiek is met een periode van 2π (zie [sheet 12](#)) kunnen we integreren over elk gewenst interval van 2π . In dit geval is het handig om te integreren van $-\pi$ tot π :

$$h[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} H(\Omega) e^{j\Omega n} d\Omega \quad (2)$$

Het invullen van de gewenste $H(\Omega)$ levert uiteindelijk de volgende impulsresponsie op:

$$h[n] = \frac{\sin\left(\frac{2}{5}\pi n\right)}{\pi n} \quad (3)$$

Dit resultaat is in de praktijk om drie redenen niet bruikbaar:

- de waarde voor $n = 0$ is onbepaald (delen door nul is niet mogelijk);
- de overdracht is niet eindig ($h[n] \neq 0$ voor $n \in \mathbb{Z}$);
- de overdracht is niet causaal ($h[n] \neq 0$ voor $n < 0$).

We lossen deze praktische problemen één voor één op.

De impulsresponsie bij $n = 0$ is eenvoudig te bepalen door de IDTFT, zie [vergelijking \(1\)](#), in te vullen en apart op te lossen voor $n = 0$: In dit geval:

$$h[0] = \frac{1}{2\pi} \int_{-\pi}^{\pi} H(\Omega) d\Omega \quad (4)$$

Het invullen van de gewenste $H(\Omega)$ levert uiteindelijk op:

$$h[0] = \frac{2}{5} \quad (5)$$

Als alternatief kun je ook de limiet van [vergelijking \(6\)](#) uitrekenen voor $n \rightarrow 0$:

$$h[0] = \lim_{n \rightarrow 0} \frac{\sin\left(\frac{2}{5}\pi n\right)}{\pi n} \quad (6)$$

In dit geval kunnen we deze limiet berekenen met de [regel van l'Hôpital](#), wat leidt tot hetzelfde resultaat:

$$h[0] = \frac{2}{5} \quad (7)$$

We maken de overdracht eindig door niet alle berekende waarden voor $h[n]$ te gebruiken, maar alleen de belangrijkste (grootste), bijvoorbeeld alleen de waarden voor $-10 \leq n \leq 10$. We passen dan een zogenoemd rectangular (rechthoekig) window toe, zie [sheet 16](#). Dit komt neer op het vermenigvuldigen van $h[n]$ met $w[n]$ waarbij geldt:

$$w[n] = \begin{cases} 1 & \text{voor } -10 \leq n \leq 10 \\ 0 & \text{voor } n < -10 \wedge n > 10 \end{cases} \quad (8)$$

Als we $N + 1$ waarden van $h[n]$ gebruiken, dan noemen we dit een N -de orde filter. Dus als we alleen de waarden van $h[n]$ gebruiken voor $-10 \leq n \leq 10$ dan noemen we dit een 20ste-orde filter.

We maken tot slot de overdracht causaal door de te gebruiken waarden van $h[n]$ over $\frac{1}{2}N$ posities naar rechts te schuiven, zie [sheet 15](#). Dit levert de zogenoemde coëfficiënten van het filter: $b[n] = h[n - \frac{1}{2}N] \times w[n - \frac{1}{2}N]$. Dit filter kunnen we realiseren met de volgende convolutie:

$$y[n] = b[n] * x[n] = \sum_{k=0}^N b[k]x[n-k] \quad (9)$$

We noemen dit filter een Finite Impulse Response (FIR)-filter omdat de impulsresponsie van dit filter eindig is. Als je dit filter als LTI-systeem implementeert, dan levert dit een systeem zonder terugkoppeling op. Een FIR-filter is dus per definitie stabiel.

Het is nu natuurlijk interessant om te bekijken wat het toepassen van het window voor effect heeft op het frequentiespectrum van het filter. Dit spectrum kan gevonden worden door de Discrete-Time Fourier Transform (DTFT) toe te passen op de uiteindelijke impulsresponsie van het filter ($b[n]$). Het resultaat van deze transformatie is weergegeven op [sheet 17](#). Je ziet:

- dat het filter minder stijl is, dan het ideale filter;
- dat er rimpels ontstaat in de pass-band;
- dat er rimpels ontstaan in de stop-band, die zijlobben genoemd worden.

In [sheet 18](#) en [19](#) zie je dat het filter voor hoge waarden van N steeds meer op het ideale filter gaat lijken.

Het door ons toegepaste rectangular window veroorzaakt rimpels (zijlobben) in het frequentiespectrum. Er zijn andere windows bedacht die de berekende $h[n]$ niet plots afkappen, maar geleidelijk af laten nemen. Op [sheet 20](#) is een voorbeeld gegeven van een zogenoemd Hamming window. Het resulterende filter is minder stijl, maar heeft ook minder last van zijlobben. De tool FilterDesigner die onderdeel is van MATLAB kan je helpen om tot een juiste keuze van een window te komen. Dit gaan we in kwartaal 4 in de praktijk toepassen.

Maak oefenopgaven 3.5.3. Als extra verdieping kun je ook [oefenopgaven 3.5.4](#) maken. De [antwoorden](#) en [uitwerkingen](#) staan op de wiki.

De in les 1 behandelde stof kun je terugvinden in de volgende paragrafen van het boek¹:

§5.1 Het ontwerpen van niet-recursieve digitale filters; Inleiding

§5.3.1 De Fourier-transformatiemethode; De basis van de methode

§5.3.2 Afkapping en tijdvensters: rechthoekige en driehoekige vensters

Les 2

De vorige les heb je geleerd om een filter te ontwerpen met behulp van de Inverse Discrete-Time Fourier Transform. Dit leverde een N^{de} -orde filter op met een impulsresponsie die $N + 1$ samples duurt. Het outputsignaal van dit filter kan als volgt worden berekend:

$$y[n] = h[n] * x[n] = \sum_{k=0}^N h[k] \cdot x[n-k] \quad (10)$$

Dit kan worden gedaan met het volgende LTI-systeem:

$$y[n] = \sum_{k=0}^N b_k \cdot x[n-k] \text{ met } b_k = h[k] \quad (11)$$

¹ P.A. Lynn. *Inleiding Digitale Signaalbewerking*. 2de ed. ThiemeMeulenhoff bv, 2004. ISBN: 9-789055-744480

Op [sheet 26](#) is dit LTI-systeem getekend. Je ziet dat er in dit filter geen terugkoppeling wordt toegepast. De frequentieoverdracht van dit filter is:

$$H(\Omega) = \frac{Y(\Omega)}{X(\Omega)} = \frac{\sum_{k=0}^N b_k e^{-jk\Omega}}{1} \quad (12)$$

We noemen zo'n filter een Finite Impulse Response filter (FIR-filter) omdat de impulsresponsie eindig is.

Als we wel terugkoppeling toepassen ontstaat een zogenoemd Infinite Impulse Response filter (IIR-filter). Op [sheet 27](#) worden de verschillen tussen FIR- en IIR-filters besproken. In [tabel 1](#) wordt een en ander samengevat. Bij filters met een lineaire fasedraaiing is de vertraging door het filter voor elke frequentiecomponent gelijk. Dit zorgt ervoor dat de originele vorm van het signaal behouden blijft. Deze eigenschap is bijvoorbeeld essentieel bij het bewerken van een ECG-sigitaal², omdat eventuele vervorming die door het filter worden geïntroduceerd, verkeerd kunnen worden geïnterpreteerd als hartafwijkingen. IIR-filters zijn vaak effectiever dan FIR-filters omdat er meestal minder coëfficiënten (b_k 's en a_k 's) nodig zijn om een gelijkwaardige frequentieresponsie te creëren. Dit leidt tot:

- een reductie in het rekenwerk want bij het berekenen van één outputwaarde moet voor elke coëfficiënt een vermenigvuldiging en een optelling uitgevoerd worden;
- minder vertraging tussen het input- en het outputsignaal want er zijn minder oude in- en outputs nodig om een nieuwe outputwaarde te berekenen als er minder coëfficiënten gebruikt worden.

Deze les ga je een methode leren om IIR-filters te ontwerpen. Volgende week bespreken we nog een derde manier om digitale filters te ontwerpen.

Op [sheet 28 en verder](#) worden het Z-domein en de Z-transformatie geïntroduceerd. Praktisch gezien is de Z-transformatie gelijk aan de Discrete-Time Fourier Transform (DTFT) waarbij $e^{j\Omega}$ gelijk gesteld is aan de complexe variabele z , zie [formuleblad](#):

$$X(z) = \sum_{n=0}^{\infty} x[n] \cdot z^{-n} \quad (13)$$

In [tabel 2](#) is het verband tussen f , Ω en z gegeven. Zorg dat je dit snapt en kunt toepassen.

² ECG staat voor elektrocardiogram, zie eventueel <https://nl.wikipedia.org/wiki/Elektrocardiogram>.

³ Zie eventueel https://en.wikipedia.org/wiki/Bessel_filter.

Tabel 1: De verschillen tussen FIR- en IIR-filters

Eigenschap	FIR	IIR
Impulsresponsie	Eindig (Finit)	Oneindig (Infiniet)
Rekursief	Nee	Ja
Terugkoppeling	Nee	Ja
Noemer van $H(\Omega)$	Onafhankelijk van Ω	Afhankelijk van Ω
Effectiviteit	Minder	Meer
Stabiliteit	Altijd stabiel	Mogelijk instabiel
Lineaire fase	Ja, als coëfficiënten symmetrisch zijn	Nee, kan wel benaderd worden, b.v. Bessel-filters ³

Tabel 2: het verband tussen f , Ω en z

f	Ω	z	opmerking
0	0	1	gelijkspanningscomponent
$\vdots$	$\vdots$	$\vdots$	
$\frac{1}{4}f_s$	$\frac{1}{2}\pi$	j	
$\vdots$	$\vdots$	$\vdots$	
$\frac{1}{2}f_s$	π	-1	Niquist-frequentie
$\vdots$	$\vdots$	$\vdots$	
$\frac{3}{4}f_s$	$1\frac{1}{2}\pi$	-j	
$\vdots$	$\vdots$	$\vdots$	
f_s	2π	1	samplefrequentie

Als de frequentie f loopt van 0 naar $\frac{1}{2}f_s$ dan loopt z over de eenheidscirkel in het complexe vlak van 1, tegen de klok in, naar -1.

Vaak is het eenvoudiger om te rekenen in het Z-domein dan in het Fourier-frequentiedomein (Ω). De overdracht van een LTI-systeem in het Z-domein, $H(z)$, is te schrijven als een breuk:

$$H(z) = \frac{T(z)}{N(z)} \quad (14)$$

De versterking is het kleinst (nul) als de teller $T(z)$ nul wordt. De (complexe) waarden van z waarbij dit gebeurt noemen we de *nulpunten* van de overdracht. De versterking is het grootst als de noemer $N(z)$ (bijna) nul wordt. De (complexe) waarden van z waarbij dit gebeurt noemen we de *polen* van de overdracht.

Bij een FIR-filter is de noemer van $H(z)$ constant en dus onafhankelijk van z . Een FIR-filter heeft dus alleen nulpunten en *geen* polen.

We kunnen de overdracht van een LTI-systeem dus ook schrijven als:

$$H(z) = K \frac{(z - z_1)(z - z_2)(z - z_3) \cdots}{(z - p_1)(z - p_2)(z - p_3) \cdots} \quad (15)$$

Waarbij K constant is, $z_1, z_2, z_3, \dots$ de nulpunten zijn en $p_1, p_2, p_3, \dots$ de polen zijn van het systeem.

In [sheet 33](#) en [34](#) wordt het effect van één pool op de stabiliteit en de impulsresponsie bekeken. Op [sheet 35](#) wordt uitgelegd dat een systeem met slechts één pool een onnodige vertraging heeft van één sampleperiode (T_s). Dit kunnen we voorkomen door een nulpunt toe te voegen in de oorsprong ($z_1 = 0$). Op [sheet 36](#) wordt het effect van één pool met een nulpunt in de oorsprong bekeken.

In [sheet 37](#) en [38](#) wordt het effect van twee polen op de stabiliteit en de impulsresponsie bekeken. De overdracht van dit systeem in het Z-domein is:

$$H(z) = \frac{1}{(z - p_1)(z - p_2)} \quad (16)$$

Je kunt de differentievergelijking van het corresponderende LTI-systeem vinden door de noemer van de overdracht uit te vermenigvuldigen, de teller en de noemer van de overdracht te vermenigvuldigen met z^{-2} , de overdracht daarna te schrijven als $Y(z) = \dots$ en ten slotte terug te transformeren naar het tijddomein:

$$H(z) = \frac{1}{z^2 - z(p_1 + p_2) + p_1 p_2} = \frac{z^{-2}}{1 - (p_1 + p_2)z^{-1} + p_1 p_2 z^{-2}} \quad (17)$$

$$H(z) = \frac{Y(z)}{X(z)} = \frac{z^{-2}}{1 - (p_1 + p_2)z^{-1} + p_1 p_2 z^{-2}} \quad (18)$$

$$Y(z) - (p_1 + p_2)Y(z)z^{-1} + p_1 p_2 Y(z)z^{-2} = X(z)z^{-2} \quad (19)$$

$$Y(z) = (p_1 + p_2)Y(z)z^{-1} - p_1p_2Y(z)z^{-2} + X(z)z^{-2} \quad (20)$$

$$y[n] = (p_1 + p_2)y[n-1] - p_1p_2y[n-2] + x[n-2] \quad (21)$$

Dit is dus een LTI-systeem met de volgende coëfficiënten:

$$a_1 = p_1 + p_2, \quad a_2 = -p_1p_2, \quad b_2 = 1 \quad (22)$$

De signalen $x[n]$ en $y[n]$ bevatten uiteraard alleen reële (en *geen* complexe) getallen. De coëfficiënten van het LTI-systeem moeten dus ook reële (en *geen* complexe) getallen zijn. De polen p_1 en p_2 kunnen echter wel complexe getallen zijn. Maar er zal altijd gelden: $p_2 = \overline{p_1}$, oftewel p_2 is de complex geconjugeerde⁴ van p_1 . Dit is eenvoudig te begrijpen. De coëfficiënten $p_1 + p_2$ en $-p_1p_2$ moeten beide reëel zijn. We kunnen p_1 schrijven als $p_1 = r_a + jr_b$ waarbij r_a en r_b reëel zijn. Ook kunnen we p_2 schrijven als $p_2 = r_c + jr_d$ waarbij r_c en r_d reëel zijn. Omdat $p_1 + p_2 = r_a + jr_b + r_c + jr_d = r_a + r_c + j(r_b + r_d)$ reëel moet zijn, moet gelden $r_d = -r_b$. Omdat $-p_1p_2 = -(r_a + jr_b)(r_c - jr_b) = -r_ar_c + jr_b(r_a - r_c) - r_b^2$ reëel moet zijn, moet gelden $r_c = r_a$. Dus als $p_1 = r_a + jr_b$, dan moet gelden $p_2 = r_a - jr_b = \overline{p_1}$.

Verder blijkt dat een LTI-systeem alleen stabiel is als $|p_k| < 1$ voor elke pool p_k . Dat wil zeggen dat alle polen binnen de eenheidscirkel in het Z-vlak moeten liggen. Als er één of meer polen op de eenheidscirkel liggen, dan is het systeem marginaal stabiel, dat wil zeggen dat de output constant is of oscilleert rondom twee waarden. Als er één of meer polen buiten de eenheidscirkel liggen, dan is het systeem instabiel.

Op [sheet 39](#) wordt uitgelegd dat een systeem met slechts twee polen een onnodige vertraging heeft van twee sampleperiode ($2T_s$). Dit kunnen we voorkomen door twee nulpunten toe te voegen in de oorsprong ($z_1 = 0$ en $z_2 = 0$), zie [sheet 39](#).

In [sheet 40](#) wordt het effect van één nulpunt op de stabiliteit en de impulsresponsie bekeken. Daarbij moet wel een pool in de oorsprong ($p_1 = 0$) worden toegevoegd omdat het systeem anders niet causaal is.

In [sheet 41](#) wordt het effect van twee nulpunten op de stabiliteit en de impulsresponsie bekeken. Er moeten twee polen in de oorsprong worden toegevoegd om het systeem causaal te maken. Dit LTI-systeem heeft de volgende coëfficiënten:

$$b_0 = 1, \quad b_1 = -(z_1 + z_2), \quad b_2 = z_1z_2 \quad (23)$$

⁴ Zie indien nodig: https://nl.wikipedia.org/wiki/Complex_geconjugeerde.

Op dezelfde manier dan hierboven voor de polen is gedaan kan worden afgeleid: $z_2 = \overline{z_2}$. Het blijkt dat dit systeem stabiel is voor elke waarde van z_k . Het maakt dus niet uit waar de nulpunten liggen in het Z-vlak.

In [sheet 42](#) wordt het effect van één pool op de frequentieresponsie bekeken. De versterking is maximaal voor $\Omega = 0$ als $p_1 > 0$ voor $\Omega = \pi$ als $p_1 < 0$. In het eerste geval is dus sprake van een laagdoorlaatfilter en in het tweede geval van een hoogdoorlaatfilter. De maximale versterking neemt toe als de pool dichterbij de eenheidscirkel ligt. Als de pool op de eenheidscirkel ligt zal de output constant zijn of oscilleren rond twee waarden en als de pool buiten de eenheidscirkel ligt zal het systeem instabiel worden.

In [sheet 43](#) wordt het effect van één nulpunt op de frequentieresponsie bekeken. De versterking is minimaal voor $\Omega = 0$ als $z_1 > 0$ voor $\Omega = \pi$ als $z_1 < 0$. In het eerste geval is dus sprake van een hoogdoorlaatfilter en in het tweede geval van een laagdoorlaatfilter. De minimale versterking neemt toe als het nulpunt dichterbij de eenheidscirkel ligt. Als het nulpunt op de eenheidscirkel ligt zal de output bij die waarde van Ω nul zijn.

Als het LTI-systeem meerdere polen en nulpunten heeft, dan is het een lastige rekenklus om de frequentieoverdracht te bepalen. Gelukkig beschikken we over allerlei computerprogramma's die ons daarbij kunnen helpen:

- Een eenvoudig MATLAB-script is beschikbaar op de wiki: [zpgui.m](#) en [plot_z_surface.m](#);
- Een uitgebreid programma is te vinden op: <http://www.micromodeler.com/dsp/>.

Gebruik een van deze programma's om te kijken wat er gebeurt met de frequentieresponsie als je polen en nulpunten in het Z-vlak verplaatst en toevoegt.

We kunnen dus een digitaal filter ontwerpen door polen en nulpunten te plaatsen in het Z-domein:

- Een pool *versterkt* signalen met frequenties in de buurt van de hoek die bij deze pool hoort.
- Een nulpunt *onderdrukt* signalen met frequenties in de buurt van de hoek die bij dit nulpunt hoort.
- Hoe dichterbij een pool of nulpunt bij de eenheidscirkel komt, hoe *selectiever* de frequentie wordt waarop de pool/nulpunt opereert.

Als we alleen nulpunten gebruiken krijgen we een FIR-filter. Als we polen gebruiken krijgen we een IIR-filter.

We hebben hierboven al gezien dat complexe polen en nulpunten altijd in complex-geconjugeerde paren voorkomen (ook wel complex-toegevoegde paren genoemd). Dit wordt op [sheet 46](#) op een andere manier nogmaals bewezen.

Als de frequentie f loopt van 0 naar $\frac{1}{2}f_s$ dan loopt z over de eenheidscirkel in het complexe vlak van 1, tegen de klok in, naar -1. Als de frequentie f loopt van 0 naar $-\frac{1}{2}f_s$ dan loopt z over de eenheidscirkel in het complexe vlak van 1, met de klok mee, naar -1. Als complexe polen of nulpunten alleen in complex-toegevoegde paren voorkomen, dan is de overdracht van 0 naar $\frac{1}{2}f_s$ gelijk aan die van 0 naar $-\frac{1}{2}f_s$ zodat de overdracht gespiegeld is rondom $f = 0$ en periodiek is over $f = f_s$. Deze eigenschappen van het spectrum van een discreet signaal hebben we in [week 1](#) al gezien.

Op [sheet 47](#) wordt de Z-getransformeerde van de eenheidspuls $\delta[n]$ en van de eenheidstap $u[n]$ bepaald. Deze transformatieparen staan ook op het [formuleblad](#). Om de Z-getransformeerde van $u[n]$ te kunnen bepalen moet je bekend zijn met de reeksontwikkeling van de geometrische reeks (ook wel meetkundige reeks genoemd):

$$\sum_{n=0}^{\infty} a^n = \frac{1}{1-a} \quad (24)$$

Ook deze formule staat op het [formuleblad](#). Als je $a = z^{-1}$ invult dan vind je de Z-getransformeerde van $u[n]$:

$$\sum_{n=0}^{\infty} z^{-n} = \frac{1}{1-z^{-1}} = \frac{z}{z-1} \quad (25)$$

Op [sheet 48](#) worden de belangrijkste eigenschappen van de Z-transformatie benoemd:

- Naar voren schuiven in de tijd met n_0 samples, is vermenigvuldigen met z^{-n_0} in het Z-domein.
- Convolutie in het tijddomein is vermenigvuldigen in het Z-domein.
- De Z-transformatie is een lineaire transformatie.

Op [sheet 49](#) wordt een voorbeeld gegeven en op [sheet 50](#) staat een vergelijkbare opgave. **Maak deze opgave.** De [uitwerking](#) staat op de wiki.

Als je een filter hebben ontworpen door polen en nulpunten in het Z-vlak te plaatsen, dan moet je de resulterende $H(z)$ terugtransformeren naar het tijddomein ($h[n]$) zodat je dit filter als een LTI-systeem kunt implementeren. Dit doe je door de inverse Z-transformatie te

gebruiken. Je maakt daarbij gebruik van bekende transformaties, zie [formuleblad](#). Daarbij pas je wiskundige technieken zoals breuksplitsen toe om de te transformeren formule in een gewenste vorm om te zetten.

Op [sheet 52](#) wordt een voorbeeld gegeven. **Voer zelf de breuksplitsing uit.** De [uitwerking](#) staat op de wiki.

Maak opgave 4.2, 4.6b, 4.8a t/m c, 4.9b, 4.11a en c uit het boek. De [antwoorden](#) en [uitwerkingen](#) staan op de wiki. Maak eventueel ook [oefenopgaven 3.5.5, 3.5.6A, 3.5.7A en C](#). Je kunt deze vragen ook gebruik om te oefenen voor de toets. De [antwoorden](#) en [uitwerkingen](#) staan op de wiki.

De in les 2 behandelde stof kun je terugvinden in de volgende paragraaf van het boek:

§4.1 Analyse in het frequentiedomein: de Z-transformatie; Inleiding

§4.2 Definitie en eigenschappen van de Z-transformatie

§4.3.1 Polen en nulpunten in het Z-vlak; Beschrijving van signalen en systemen met polen en nulpunten

§4.3.2 Geometrische evaluatie van de Fouriertransformatie in het Z-vlak

Extra verdieping les 2

Het niveau van deze verdiepende opdrachten ligt iets boven het niveau dat op de toets gevraagd wordt. Toch is het zeker zinvol om de volgende opdrachten uit het boek te maken: [4.1, 4.6a, 4.8d, 4.9a, 4.11b en 4.12](#). De [antwoorden](#) en [uitwerkingen](#) staan op de wiki.

Extra oefenopgaven week 5

Extra oefenopgaven over de stof van week 5 vind je hier: [Oefenopgaven_WK5.pdf](#). De bijbehorende antwoorden vind je hier: [Oefenopgaven_WK5_antwoorden.pdf](#). De uitwerkingen vind je hier: [Oefenopgaven_WK5_uitwerkingen.pdf](#).

Oefenopgaven 3.5.1 en 3.5.2 zijn bedoeld om de stof van week 3 te herhalen. Oefenopgaven 3.5.3 en 3.5.4 zijn bedoeld om de stof van les 1 te oefenen (3.5.4 is een verdiepende opgave). Oefenopgaven 3.5.5 en 3.5.7 zijn bedoeld om de stof van les 2 te oefenen (3.5.6B en C en 3.5.7B zijn verdiepende opgaven).